Δομές Δεδομένων-Εργασία 3

Πλαβός Δημοσθένης 3200156

Contains:

Στη συνάρτηση αυτή, επιστρέφω true αν το τετράγωνο περιέχει το σημείο p τύπου Point. Περιέχεται όταν η συντεταγμένη x του σημείου έχει τιμή στο πεδίο τιμών [xmin,xmax] του rectangle και η συντεταγμένη y του σημείου έχει τιμή στο πεδίο τιμών [ymin,ymax] του rectangle.

Intersects:

Στη συνάρτηση αυτή, επιστρέφω true αν το τετράγωνο έχει κοινό σημείο με το τετράγωνο that τύπου Rectangle. Βρίσκω αρχικά σε ποιες περιπτώσεις δεν έχουν κοινό σημείο. Η πρώτη περίπτωση είναι το ένα τετράγωνο να είναι αριστερά από το άλλο, δηλαδή το xmax του ενός να είναι μικρότερο από το xmin του άλλου. Η δεύτερη περίπτωση είναι όταν το ένα βρίσκεται πάνω από το άλλο, δηλαδή το ymin του ενός να είναι μικρότερο από του άλλου.

distanceTo της κλάσης Rectangle:

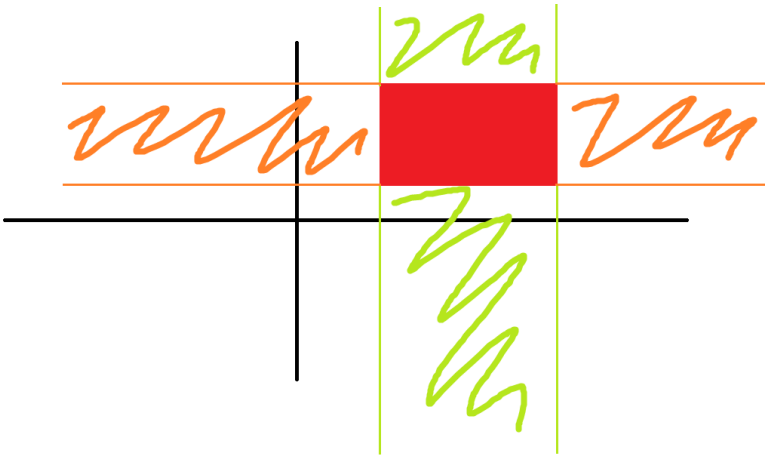
Χρησιμοποιώ τη συνάρτηση αυτή για να επιστρέψω την απόσταση του σημείου p τύπου Point με το τετράγωνο. Η απόσταση αυτή πρέπει να είναι η κοντινότερη που είναι δυνατόν. Άρα πρέπει να βρω το σημείο του τετραγώνου που είναι πιο κοντινό στο p. Για τον σκοπό αυτό, δημιουργώ τη συνάρτηση VresSimeio η οποία λειτουργεί με την εξής φιλοσοφία: Έστω το κοντινότερο σημείο που θέλω να βρω A(Xa,Ya) και P(Xp,Yp) το σημείο που δίνει ο χρήστης.Τότε,

Περίπτωση 1: Αν το σημείο A ανήκει στο κομμάτι του επιπέδου που βρίσκεται στο [xmin,xmax] (λαχανί σκιασμένη περιοχή) , τότε το κοντινότερο σημείο είναι η προβολή του p στο Rectangle. Άρα το Xa=Xp και για το Ya βρίσκω τις αποστάσεις των (α) ymin με Ya (β) ymax με Ya και όποια είναι μικρότερη μου δίνει και το σημείο.

Περίπτωση 2: Αν το σημείο Α ανήκει στο κομμάτι του επιπέδου που βρίσκεται στο [ymin,ymax], (πορτοκαλί σκιασμένη περιοχή), τότε πάλι το κοντινότερο σημείο είναι η προβολή του p στο rectangle. Ομοίως, Ya=Yp και για το Xa συγκρίνω τις αποστάσεις Xa με Xmin, Xa με Xmax.

Περίπτωση 3: Αν το σημείο Α ανήκει στο υπόλοιπο επίπεδο, τότε η πιο κοντινή απόσταση είναι μία γωνία του rectangle αναλόγως που ακριβώς βρίσκεται το p. Βρίσκω λοιπόν, ποια γωνία είναι η πιο κοντινή και έχω το σημείο Α.

Αφού έχω βρει το σημείο Α, χρησιμοποιώ την distanceTo της κλάσης Point μεταξύ των σημείων P και A για να βρω την απόσταση.



rangeSearch:

Με τη συνάρτηση αυτή θέλω να βρω τα κοινά σημεία του δέντρου με το τετράγωνο που δίνει ο χρήστης. Αυτό θα γίνει με αναδρομική κλήση των δεξιών και αριστερών υποδέντρων, ξεκινώντας από το head.

Κάθε κόμβος του δέντρου αντιστοιχεί σε ένα rectangle.Για να αξιοποιήσω το πλεονέκτημα των 2D-trees, αν το rectangle του κόμβου στον οποίο βρίσκομαι δεν έχει κοινά σημεία με το τετράγωνο που δίνει ο χρήστης, δεν συνεχίζω την αναζήτηση στα υποδέντρα του κόμβου αυτού.

Για να βρω το rectangle που αντιστοιχεί κάθε κόμβος, κάνω τη διαδικασία αυτή εντός της συνάρτησης insert. Όταν εισάγω έναν κόμβο, διακρίνω περιπτώσεις αν η συντεταγμένη η οποία παίζει ρόλο στην εισαγωγή του είναι η χ ή η υ εναλλάξ, και περιπτώσεις για το αν εισάγεται ως δεξιό ή αριστερό παιδί του πατέρα. Σε αυτές τις 4 περιπτώσεις δημιουργώ το αντικείμενο τύπου Rectangle αξιοποιώντας και τις συντεταγμένες του πατέρα.

nearestNeighbor:

Με τη συνάρτηση αυτή θέλω να βρω το πιο κοντινό σημείο του δέντρου στο σημείο που δίνει ο χρήστης.

Χρησιμοποιώ τη squareDistanceTo καθώς διασχίζω τους κόμβους του δέντρου αναδρομικά. Ομοίως με την rangeSearch, αξιοποιώ το rectangle που αντιστοιχεί σε κάθε κόμβο έτσι ώστε αν η απόσταση του p από τον κόμβο είναι >= από την απόσταση του p από το τετράγωνο, να μην συνεχίζω τις αναδρομές στα υποδέντρα του κόμβου.